

**Amendments to the Claims:**

1 – 125 (Canceled)

126. (Currently Amended) A method comprising:

receiving, through a network, a form authored ~~by a user~~ using a form authoring language, the form containing one or more input fields;  
parsing the received form to identify the input fields contained in the received form;

providing a graphical user interface ~~to the user~~ to allow ~~the user to identify identification of~~ actions to be associated with the identified input fields upon subsequent specific submission of a specific instance the form by a third party, the provided graphical user interface being dependent on the identified input fields;  
automatically generating a program code to carry out the actions identified by the user associated with the identified input fields.

127. (Previously Presented) The method of claim 126, wherein the form authoring language is Hyper Text Markup Language (HTML).

128. (Previously Presented) The method of claim 126, wherein the generated program code is a Common Gateway Interface (CGI) program.

129. (Previously Presented) The method of claim 126, further comprising associating the generated program code with the form.

130. (Currently Amended) The method of claim 129, further comprising:

receiving a specific submission of the form from a third party, including data from the input fields; and  
executing the program code on the received data.

131. (Previously Presented) The method of claim 126, further comprising automatically determining whether the generated program code is consistent with the form; and generating an alert if the generated program code is not consistent with the form.

132. (Currently Amended) A server comprising:

a communications device connected to a network to receive a form from a client connected to the network, the form authored ~~by a user~~ using a form authoring language and containing one or more input fields;

a memory coupled to the communications device to store the received form;  
a parser module coupled to the memory to parse the received form to identify the input fields contained in the received form;

a configurer module coupled to the parser module to create a graphical user interface based on the input fields identified by the parser module, and to provide the graphical user interface ~~to the user~~ using the communications device to allow the ~~user to~~ identify identification of actions to be associated with the identified input fields upon subsequent submission of a specific instance of the form by a third party;

a code generator module coupled to the configurer module to automatically generate a program code to carry out the actions identified ~~by the user~~.

133. (Previously Presented) The server of claim 132, further comprising a processor, wherein the parser module, the configurer module, and the code generator module are all implemented using the processor.

134. (Previously Presented) The server of claim 132, wherein the form authoring language is Hyper Text Markup Language (HTML).

135. (Previously Presented) The server of claim 132, wherein the generated program code is a Common Gateway Interface (CGI) program.

136. (Previously Presented) The server of claim 132, further comprising a consistency module to automatically determine whether the generated program code is consistent with the form, wherein the consistency module generates an alert if the program code is not consistent with the form.

137. (Currently Amended) A machine-readable medium containing data representing instructions that, when executed by a processor, cause the processor to perform operations comprising:

receiving, through a network, a form authored ~~by a user~~ using a form authoring language, the form containing one or more input fields;  
parsing the received form to identify the input fields contained in the received form;

providing a graphical user interface to the user to allow ~~the user to identify~~ identification of actions to be associated with the identified input fields upon subsequent submission of a specific instance of the form by a third party, the provided user graphical user interface being dependent on the identified input fields;

automatically generating a program code to carry out the actions ~~identified by the user~~ associated with the identified input fields.

138. (Previously Presented) The machine-readable medium of claim 137, wherein the form authoring language is Hyper Text Markup Language (HTML).

139. (Previously Presented) The machine-readable medium of claim 137, wherein the generated program code is a Common Gateway Interface (CGI) program.

140. (Previously Presented) The machine-readable medium of claim 137, wherein the instructions further cause the processor to perform operations comprising associating the generated program code with the form.

141. (Currently Amended) The machine-readable medium of claim 140, wherein the instructions further cause the processor to perform operations comprising:

receiving a specific instance of the form from a third party, including data from the input fields; and

executing the program code on the received data.

142. (Previously Presented) The machine-readable medium of claim 137, wherein the instructions further cause the processor perform operations comprising automatically determining whether the generated program code is consistent with the form; and

generating an alert if the generated program code is not consistent with the form.

143. (New) A method comprising:

receiving, through a network, a form authored using a form authoring language, the form containing one or more input fields;

parsing the received form to identify the input fields contained in the received form;

providing a graphical user interface to allow identification of actions to be associated with the identified input fields upon subsequent submission of a specific instance of the form by a third party, the provided user graphical user interface being dependent on the identified input fields;

automatically generating a program code to carry out the actions associated with the identified input fields;

receiving a specific instance of the form, including data from the input fields; and

executing the program code on the received data.

144. (New) The method of claim 143 wherein the actions to be performed by the program code comprises:

validating one or more form elements included in the form.

145..(New) The method of claim 143 wherein the actions to be performed by the program code comprises:

generating one or more quantities based upon submitted values with respect to one or more form elements contained in the received specific instance of the form.

146. (New) The method of claim 143 wherein the actions to be performed by the program code comprises:

generating one or more licenses in response to the submission of the specific instance of the form.

147. (New) The method of claim 143 wherein the actions to be performed by the program code comprises:

generating one or more cookies for each user who submits a specific instance of the form for processing.

148. (New) The method of claim 143 wherein the actions to be performed by the program code comprises:

emailing the contents of the specific instance of the form to one or more addresses specified in response to the submission of a specific instance of the form.

149. (New) The method of claim 143 wherein the actions to be performed by the program code comprises:

sending an email message to a user who submits a specific instance of the form.

150. (New) The method of claim 143 wherein the actions to be performed by the program code comprises:

generating one or more responses upon receipt of the specific instance of the form.

151. (New) The method of claim 143 wherein the actions to be performed by the program code comprises:

preserving a state of one or more fields in the specific instance of the form and passing the state of the one or more fields in the form to a second specific instance of the form.

152. (New) The method of claim 143 wherein the actions to be performed by the program code comprises:

logging values for one or more form elements included in the specific instance of the form.

153. (New) The method of claim 152 wherein the values are stored in a database.

154. (New) The method of claim 153 wherein the values in each submission are stored in a single row of a table, different submissions corresponding to different rows of the table.

155. (New) A machine-readable medium comprising instructions which, when executed by a machine, cause the machine to perform operations comprising:  
allowing an author of a first form who is located at a first location to remotely configure a first program's functions at a second location, the first program to perform its corresponding functions in processing submissions of the first form from an end user, said allowing comprising:

receiving the first form from the author at a first server;  
parsing the first form to extract specification information regarding form elements included in the first form;

creating a representation of the form elements based upon the specification information extracted from the first form;  
obtaining configuration information regarding the form elements of the first form; and  
configuring the first program's functions using the configuration information provided.

156. (New) The machine-readable medium of claim 155 wherein the actions to be performed by the first program comprise a field validation function that includes validating one or more form elements included in the first form.

157. (New) The machine-readable medium of claim 155 wherein the actions to be performed by the first program include a derived quantity generation function that includes generating one or more quantities based upon a user's submitted values with respect to one or more form elements contained in the first form.

158. (New) The machine-readable medium of claim 155 wherein the actions to be performed by the first program include a license generation function that includes generating one or more licenses in response to the first form's submission.

159. (New) The machine-readable medium of claim 155 wherein the actions to be performed by the first program include a cookie generation function that includes generating one or more cookies for each user who submits the first form for processing.

160. (New) The machine-readable medium of claim 155 wherein the actions to be performed by the first program include a form emailing function that includes emailing the contents of the first form's submission to one or more addresses specified by the first author in response to the submission of the first form by a user.

161. (New) The machine-readable medium of claim 155 wherein the actions to be performed by the first program include an email-to-the-user function that includes sending an email message to a user who submits the first form.

162. (New) The machine-readable medium of claim 155 wherein the actions to be performed by the first program include a form response function that includes generating one or more responses upon receipt of the first form's submission from a user.

163. (New) The machine-readable medium of claim 155 wherein the actions to be performed by the first program include a form threading function that includes preserving a state of one or more fields in the first form and passing the state of the one or more fields in the first form to a second form.

164. (New) The machine-readable medium of claim 155 wherein the actions to be performed by the first program include a form logging function that includes logging a user's submitted values for one or more form elements included in the first form.

165. (New) The machine-readable medium of claim 164 wherein the submitted values are stored in a database.

166. (New) The machine-readable medium of claim 165 wherein the values in each submission are stored in a single row of a table, different submissions corresponding to different rows of the table.

167. (New) The machine-readable medium of claim 155 further comprising:  
providing the first author with a capability to manage data relating to the submissions of the first form by one or more users.